5П

УДК 62-50:15

***Н.Б. Паклин*[1]*, С.В. Уланов*[2]*, С.В. Царьков*[1]**
[1]Рязанский филиал МЭСИ, г. Рязань, Россия
[2]Сарапульский политехнический институт, г. Сарапул, Россия
pnb@inbox.ru

Построение классификаторов на несбалансированных выборках на примере кредитного скоринга

В статье рассмотрена проблема построения эффективных бинарных классификаторов в условиях, когда классы сильно несбалансированы. Показана их связь с издержками классификации, рассмотрены стратегии борьбы с несбалансированностью и проведены эксперименты на кредитных историях российских банков.

Введение

При решении многих практических задач методами машинного обучения исследователи сталкиваются с тем, что в обучающей выборке присутствует несбалансированность классов, то есть классы представлены неравномерно (imbalanced dataset). В частности, эта проблема актуальна при построении бинарного классификатора при решении задачи кредитного скоринга, так как доля «плохих» заемщиков крайне редко превышает 15%, а в большинстве случаев находится на уровне 3-6%. Например, при построении дерева классифицирующих правил [1] на такой обучающей выборке может оказаться, что результирующая модель содержит крайне мало правил или вовсе «пустое» дерево.

Несбалансированность классов оборачивается и другими трудностями. Классификаторы, построенные на основе выборки, в которой репрезентативность классов несбалансирована, имеют в процессе практического использования склонность с большей вероятностью относить новые наблюдения к классам, представленным большим числом обучающих примеров. Поставленная проблема усложняется существенным различием издержек ошибок классификации.

В случае если неблагонадежный клиент был распознан классификатором как «хороший», то имеет место ошибка первого рода. Также возникают ситуации, когда благонадежный клиент распознан в качестве «плохого», такая ситуация называется ошибкой второго рода. Издержки классификации в каждом случае существенно отличаются. Совершенно очевидно, что убыток от выдачи кредита неблагонадежному клиенту во много раз превышает упущенную прибыль при отказе «хорошему». То есть самым интересным оказывается наименее представленный класс.

Кредитный скоринг – это не единственная предметная область, где актуальна указанная проблема несбалансированности [2]. При обнаружении мошенничеств медицинской диагностики также наблюдается несбалансированность классов с существенным различием издержек ошибочной классификации.

Цель данной работы заключалась в анализе существующих подходов к решению проблемы создания классификаторов на несбалансированных выборках и их апробация на задаче кредитного скоринга, что сегодня очень востребовано в банковских информационно-аналитических системах.

Постановка задачи

Введем следующие два понятия. Класс, представленный в обучающих данных меньшим числом примеров, назовем миноритарным (от англ. minority – меньшинство), а представленный бóльшим числом примеров – мажоритарным (от англ. majority – большинство).

Как известно, эффективность бинарного классификатора описывается матрицей классификации. При этом обычно миноритарный класс принимается за положительный (1), а мажоритарный – за отрицательный (0). Тогда матрица будет иметь вид, показанный на рис. 1.

Предсказанный класс			
Фактический класс		Класс «+»	Класс «–»
	Класс «+»	Истинноположительный (11)	Ложноотрицательный (10)
	Класс «–»	Ложноположительный (01)	Истинноотрицательный (00)

Рисунок 1 – Матрица классификации (случай с двумя классами)

Этой матрице будет соответствовать матрица издержек, которая показывает издержки, связанные со всеми четырьмя возможными исходами C_{11}, C_{10}, C_{01} и C_{00}. Издержки в случае правильной классификации одинаковы, поэтому величины C_{11} и C_{00} полагаются равными 0. Также в силу того, что миноритарный класс представляет больший интерес, $C_{01} < C_{10}$.

Формальная постановка задачи классификации с учетом издержек следующая. Пусть имеем задачу построения бинарного классификатора на множестве обучающих примеров $(\mathbf{X}_i, y)$, $i = 1,\ldots, n$, $\mathbf{X}_i$ – вектор признаков, y – метка класса из множества $Y = \{1,2,\ldots,J\}$. Кроме этого, предположим, что обучающая выборка была получена из множества, распределенного по некоторому вероятностному закону $P(\mathbf{X}, y)$. Тогда целью алгоритма обучения будет построение классификатора h, который делает возможным правильное распознавание произвольных примеров, распределенных по тому же закону с достаточно высокой вероятностью. Аналогично, если неправильное распознавание ведет к издержкам (или потерям), то целью обучения будет минимизация полных ожидаемых издержек C_t:

$$C_t = \sum_{(\mathbf{X},y)} P(\mathbf{X}, y) C(h(\mathbf{X}), y),$$

где $C(h(\mathbf{X}), y)$ – функция издержек, выражающая удельные потери на пример $(\mathbf{X}, y)$. Таким образом, полные издержки C_t представляют собой сумму издержек для всех классифицируемых наблюдений.

Заметим, что в обычной задаче классификации функция издержек $C(h(\mathbf{X}), y)$ равна 1 при $h(\mathbf{X}) \neq y$ и 0 – в противном случае. Классификаторы такого типа известны как минимизаторы ожидаемых издержек. На практике издержки ошибочной классификации неодинаковы для различных классов, функция издержек должна быть задана.

Пусть для классификатора h известна вероятность $P_h(i, j)$ того, что случайно выбранный пример относится к классу j, но распознается как i. Тогда ожидаемые издержки классификатора h будут равны:

$$L(h) = \sum_{i=1}^{m} \sum_{j=1}^{m} P_h(i, j) C(i, j). \qquad (1)$$

Следует отметить, что $P_h(i, j) = P(i \mid j) P(j)$, где $P(j)$ – вероятность того, что отдельный пример относится к классу j, а $P_h(i \mid j)$ – условная вероятность ошибочного отнесения примеров класса j к классу i.

Таким образом, целью задачи классификации с учетом издержек является нахождение классификатора, который минимизирует полные издержки на основе уравнения (1).

В кредитном скоринге часто в качестве выходной переменной большой интерес представляет скоринговый балл, $R = R(\mathbf{X})$, – непрерывное значение, лежащее в промежутке [0, 1]. Значение R в данном случае можно рассматривать как некоторую оценочную вероятность того, что клиент с вектором признаков **X** принадлежит к классу 1. Результат классификации в таком случае можно изменять путем повышения или понижения порога отсечения t. Если ошибка классификации клиента из класса 1 к классу 0 в r раз важнее (например, если издержки высоки), то, согласно правилу Байеса, минимальные издержки достигаются при $t = (1 + r)^{-1}$. К сожалению, этот способ применим лишь для классификаторов, которые на выходе дают возможность варьировать параметром t. Например, логистическая регрессия или простой байесовский классификатор. А такие эффективные нелинейные методы, как нейронные сети и машины опорных векторов, не обладают этой возможностью.

В связи с этим были разработаны альтернативные подходы для решения проблемы построения эффективных бинарных классификаторов, основанные на изменении пропорций классов целевой переменной в выборке (специальные типы *сэмплинга*).

Изменение репрезентативности классов

Данный подход использует сэмплинг для изменения распределения классов и называется *восстановлением равновесия* (rebalancing) с целью получения более сбалансированного обучающего множества [3], [4]. К основным методам сэмплинга относят *выборку с дублированием миноритарного класса* (oversampling) и *выборку с удалением примеров мажоритарного класса* (undersampling). В первой ситуации случайным образом выбирается n записей миноритарного класса и их полностью копируют, во второй – удаляют k записей мажоритарного класса.

Возникает вопрос: на сколько конкретно нужно увеличивать число примеров миноритарного (редкого) класса или удалять из мажоритарного класса? Ответ на этот вопрос дает следующее утверждение, связывающее правило Байеса для определения оптимального порога и число примеров обоих классов [5]: при использовании в классификаторе порога отсечения 0,5 и при условии, что $C_{00} = C_{11} = 0$, число примеров миноритарного класса нужно увеличить в C_{10} / C_{01} раз.

Данное утверждение позволяет понять, как нужно изменить соотношение примеров в обучающем множестве, чтобы это было равносильно изменению порога отсечения для принятия решения о принадлежности к классу. Можно пойти другим путем – уменьшить число записей мажоритарного класса в C_{10} / C_{01} раз.

Поясним утверждение на примере. Пусть имеется обучающее множество с кредитными историями заемщиков, в котором 900 записей о хороших заемщиках и 100 – о плохих (редкий класс). Пусть известно, что отношение издержек равно 5:1. Тогда по правилу Байеса оптимальным порогом в логистической регрессии будет величина $t > 1 / (1+5) = 0,167$ при условии, что мы не производим изменение баланса классов и за положительный исход принимаем плохого клиента. Если мы оставляем порог, равный 0,5, то согласно процедуре *oversampling* необходимо продублировать еще 400 записей, относящихся к плохим клиентам (общий объем выборки составит 1000 + 400 = 1400 примеров), а согласно процедуре *undersampling* – уменьшить число хороших до 900 / 5 = 180 клиентам (общий объем выборки составит 180 + 100 = 280 примеров).

Помимо основных методов сэмплинга существуют и специальные. Так, главная идея одностороннего сэмплинга (one-side sampling) заключается в нахождении и последующем удалении из набора данных таких записей мажоритарного класса, которые зашумляют выборку. Для этого проделывают следующие шаги [6].

1. Пусть S – исходный набор данных.

2. Инициализировать поднабор G, содержащий все записи миноритарного класса из S и одну случайно выбранную записи мажоритарного.

3. Классифицировать исходный набор данных по правилу одного ближайшего соседа, используя примеры из G.

4. Переместить ошибочно классифицированные примеры в поднабор.

5. Удалить каждый попавший в G мажоритарный пример i, для которого найдется такая запись k, что будет справедливо следующее условие:

$$\left[\begin{array}{l} d(\mathbf{X}_i, \mathbf{X}_k) < d(\mathbf{X}_i, \mathbf{X}_j); \\ d(\mathbf{X}_j, \mathbf{X}_k) < d(\mathbf{X}_j, \mathbf{X}_i), \end{array}\right.$$

где $d(\mathbf{X}_i,\mathbf{X}_j)$ – это расстояние между векторами признаков записей $\mathbf{X}_i$ и $\mathbf{X}_j$, j – пример из миноритарного класса.

В основе другой процедуры – специальной выборки с дублированием миноритарного класса (focused oversampling) – лежит алгоритм SMOTE [7]. Он основан на идее генерации некоторого количества искусственных примеров, которые были бы «похожи» на имеющиеся в миноритарном классе, но при этом не являлись дубликатами. Для создания нового примера находят вектор $\mathbf{d} = \mathbf{X}_b - \mathbf{X}_a$, где $\mathbf{X}_a$, $\mathbf{X}_b$ – векторы признаков «соседних» примеров a и b из миноритарного класса. Далее из $\mathbf{d}$ путем умножения каждого его элемента на случайное число в интервале (0, 1) получают $\tilde{\mathbf{d}}$. Вектор признаков нового примера получается путем сложения векторов $\mathbf{X}_a$ и $\tilde{\mathbf{d}}$. Процедура SMOTE позволяет задавать количество примеров, которое необходимо искусственно сгенерировать. Степень сходства примеров a и b можно регулировать значением k (числом ближайших соседей).

Строгих теоретических обоснований эти процедуры не имеют. Предполагается, что смещение, внесенное в обучающие данные, позволит алгоритму обучения получить модель, которая минимизирует издержки при классификации новых наблюдений. Главное преимущество сэмплинга, который изменяет равновесие классов, заключается в том, что он не требует модификации алгоритма обучения, является простой процедурой и может применяться к любым типам классификаторов. Его использование позволяет строить модели, оптимальные с точки зрения издержек классификации. Но есть недостатки. Так, выборка с удалением примеров мажоритарного класса может вызвать потерю потенциально полезной информации, которая содержится в исключаемых примерах. А «клонирование» большого числа одинаковых примеров способно привести к переобучению модели, что экспериментально доказано в работах [1-4].

Модификация алгоритма обучения

Здесь производится модификация алгоритма построения классификатора таким образом, чтобы он учитывал издержки ошибок классификации. В настоящее время для многих алгоритмов существуют такие модификации. Например, при построении дерева классифицирующих правил одним из наиболее популярных методов является использование информации об издержках неправильной классификации при выборе атрибута ветвления в каждом узле строящегося дерева. Одно из расширений алгоритма C4.5 [1] использует для выбора атрибута комбинированный критерий, учитывающий как приращение информации, так и ошибки издержек классификации. Для этого вводится функция, несущая информацию об издержках классификации. Для k- го атрибута она определяется как $ICF_k = \left(2^{\Delta I_k} - 1\right) / \left(C_{ij} + 1\right)^{\alpha}$, где $0 \le \alpha \le 1$, ΔI_k – прирост информации, связанный с разбиением по k-му атрибуту, C_{ij} – издержки, связанные с классами, примеры которых участвовали в разбиении.

Параметр α позволяет варьировать степенью «стремления» алгоритма к выбору атрибутов, с которыми связаны меньшие издержки. Если $\alpha = 0$, а $ICF = 1$, то издержки не учитываются. Если $\alpha = 1$, то имеет место максимальное влияние издержек на процесс построения дерева. Регулируя значение данного параметра, исследователь добивается оптимальной чувствительности алгоритма к издержкам классификации.

Все же многие исследователи отдают предпочтение процедурам восстановления равновесия, а не модифицированным алгоритмам обучения [1], [3], [4]. Для этого есть несколько причин. Одна из них заключается в том, что не для всех алгоритмов машинного обучения разработаны модифицированные варианты, учитывающие издержки ошибок классификации. Другая – в том, что число примеров с доминирующим классом часто избыточно, и тогда выборка с удалением примеров мажоритарного класса кажется наиболее привлекательной процедурой.

Кроме того, издержки ошибок классификации часто неизвестны, что затрудняет использование методов обучения, чувствительных к издержкам. Если информация об издержках отсутствует, то для оценки эффективности бинарного классификатора можно использовать такие методы, как графики «чувствительность – специфичность», больше известные как ROC-кривые.

Экспериментальная часть

Целью экспериментов являлось исследовать эффективность различных подходов к построению кредитных скоринговых моделей в условиях несбалансированности классов. Для этого мы использовали два набора данных с реальными кредитными историями российских банков (их описание приведено в табл. 1), причем одна из них затрагивает послекризисный период 2008 года, и три изложенных выше подхода: две процедуры сэмплинга и алгоритм построения дерева решений C5.0, учитывающий издержки классификации. Выборки, полученные при помощи сэмплинга, также подавались на вход алгоритма C5.0, но матрица издержек уже не задавалась.

Таблица 1 – Наборы данных, участвующие в эксперименте

Характеристика	Набор 1	Набор 2
Банк	Российский банк ТОП-30	Российский банк
Типы кредитов	Потребительский	Потребительский
Период выдачи кредитов	11.2006 – 05.2007	09.2008 – 2009
Объем множества	4244	944
Доля «плохих» кредитов	17%	14,5%
Число переменных	22	14

Пропорции обучающего и тестового множества составили 75% и 25% соответственно. Издержки C_{10} (за положительный исход принят «плохой» заемщик) брались равными поочередно 2, 3, 4, 6, 10, 50. Классификаторы создавались по 10 попыток для каждого отношения этих издержек, а результаты усреднялись. Они приведены на рис. 2 и рис. 3 в виде графиков зависимостей C_t от C_{01} / C_{10}.

Их анализ не позволил признать какую-либо одну стратегию обучения выигрышной, что совпадает с работой [3]. Тем не менее, на промежутке от «1:4» до «1:10», то есть когда отношение издержек ложноотрицательных к ложноположительным ошибкам варьируется от 4 до 10, что является типичной ситуацией в кредитном скоринге, лучшие результаты показывает алгоритм C5.0, а худшие – процедура *undersampling*. С ростом C_{10} этот метод отстает от других, делая его использование непригодным уже при $C_{10} > 10$.

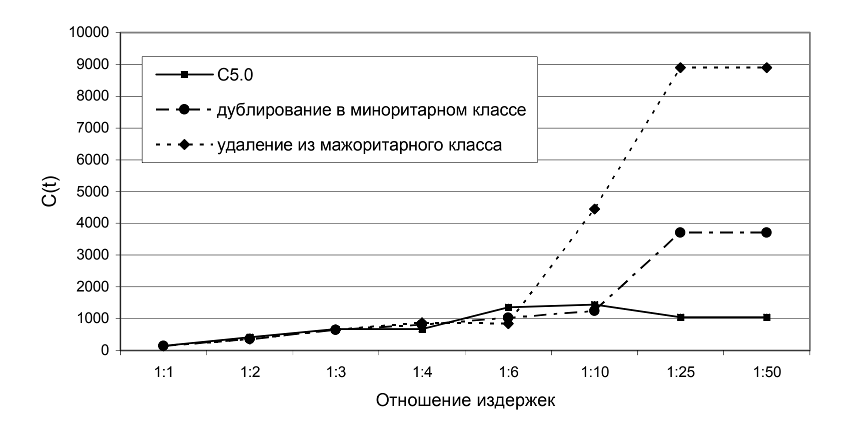


Рисунок 2 – Графики зависимостей C_t от C_{01}/C_{10} для набора № 1

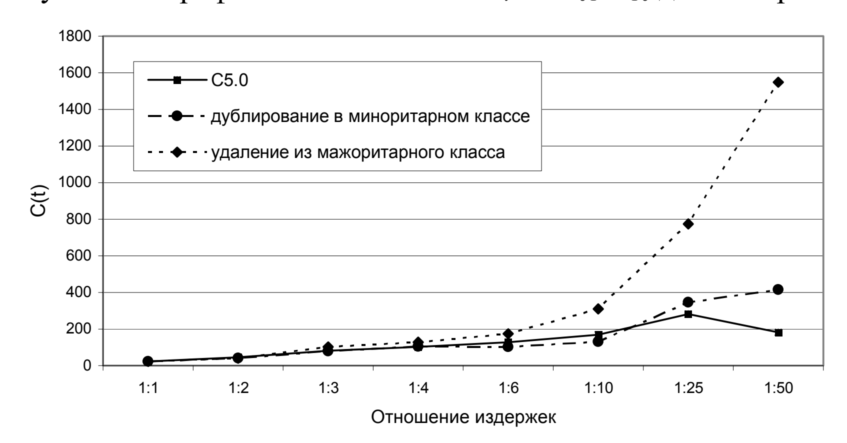


Рисунок 3 – Графики зависимостей C_t от C_{01}/C_{10} для набора № 2

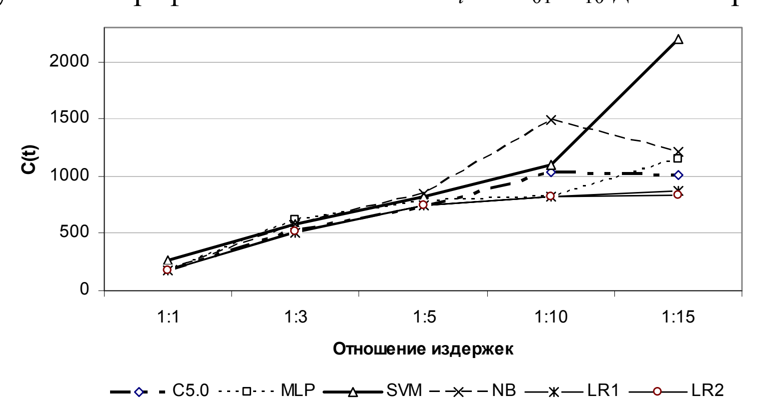


Рисунок 4 – Графики полных издержек для различных скоринговых алгоритмов в зависимости от отношения издержек, процедура *oversampling*

Далее проводились эксперименты, в которых участвовали несколько различных алгоритмов классификации (рис. 4 и рис. 5, обозначения: *LR1* – логистическая регрессия с порогом округления 0,5; *MLP* – многослойный персептрон; *SVM* – машины опорных векторов; *NB* – простой классификатор Байеса; *LR2* – логистическая регрессия с порогом, рассчитанным по правилу Байеса).

Анализ этих графиков показал, что наиболее стабильные и лучшие результаты (минимальное значение C_t) обеспечивает логистическая регрессия (обе процедуры сэмплинга), а также алгоритм дерева решений C5.0 (процедура *undersampling*). С увеличением отношения издержек наихудшие результаты демонстрирует метод машин опорных векторов.

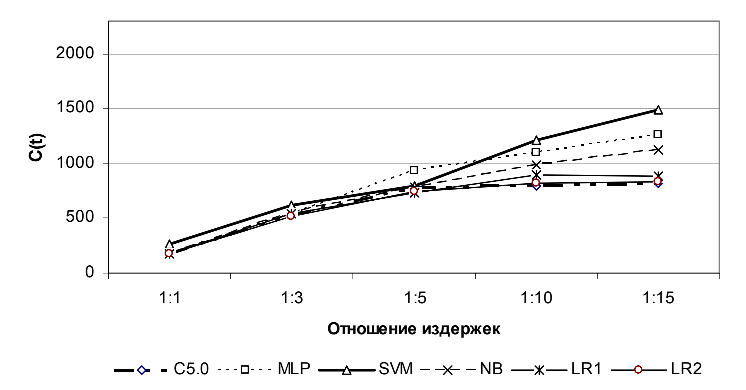


Рисунок 5 – Графики полных издержек для различных скоринговых алгоритмов в зависимости от отношения издержек, процедура *undersampling*

Выводы

При решении многих практических задач стандартные алгоритмы машинного обучения не позволяют создавать эффективные классификаторы из-за несбалансированных обучающих выборок.

Главное преимущество метода сэмплинга с восстановлением равновесия классов заключается в том, что он не требует модификации алгоритма обучения, является простой процедурой и может применяться к любым типам классификаторов.

Показано, что в кредитном скоринге при помощи метода oversampling строятся эффективные классификаторы, не уступающие другим подходам, которые обеспечивают любое соотношение ошибок I и II рода, а значит, подбор порогового скорингового балла.

Перспективным представляется исследование и сравнение метода сэмплинга SMOTE для задачи кредитного скоринга.

Литература

1. Chawla N. C4.5 and imbalanced datasets: investigating the effect of sampling method, probabilistic estimate, and decision tree structure / N. Chawla // ICML 2003 Workshop on Imbalanced Datasets.
2. Vinciotti V. Scorecard construction with unbalanced class sizes / V. Vinciotti, D.J. Hand // Journal of Iranian Statistical Society. – 2002. – Vol. 2 – P. 189-205.
3. Weiss G.M. Cost-Sensitive Learning vs. Sampling: Which is Best for Handling Unbalanced Classes with Unequal Error Costs? / G.M. Weiss, K. McCarthy, B. Zabar // Proceedings of the 2007 International Conference on Data Mining, CSREA Press, 2007. – P. 35-41.
4. McCarthy K. Does Cost-Sensitive Learning Beat Sampling for Classifying Rare Classes? / K. McCarthy, Zabar B., Weiss G.M. // Proceedings of the First International Workshop on Utility-Based Data Mining (at KDD-05), ACM Press, 2005. – P. 69-75.
5. Elchan Ch. The Foundations of Cost-Sensitive Learning / Ch. Elchan // Proc. of the 17th International Joint Conference on Artificial Intelligence, 2001. – P. 973-978.
6. Kubat M. Addressing the curse of imbalanced training sets: one-sided selection / M. Kubat, S. Matwin // In: Proc. 14th International Conference on Machine Learning, 1997. – P. 179-186.
7. SMOTE: Synthetic Minority Over-sampling Technique / N. Chawla, K. Bowyer, L. Hall, W. Kegelmeyer // Journal of Artificial Intelligence Research. – 2002. – Vol. 16. – P. 341-378.

М.Б. Паклін, С.В. Уланов, С.В. Царьков

Побудова класифікаторів на незбалансованих вибірках на прикладі кредитного скорингу

У статті розглянута проблема побудови ефективних бінарних класифікаторів в умовах, коли класи сильно незбалансовані. Показаний їх зв’язок з витратами класифікації, розглянуті стратегії боротьби з незбалансованістю та проведені експерименти на кредитних історіях російських банків.

N.B. Paklin, S.V. Ulanov, S.V. Tsarkov

Classifiers Construction Based on Imbalanced Datasets by the Example of Credit Scoring

The article discusses the problem of constructing efficient binary classifiers on imbalanced datasets. Costs of classification and strategies to win the imbalance are considered. Experiments on the credit histories of Russian banks are made.

Статья поступила в редакцию 01.07.2010.